Modeling Intuition as a System

"To explain the integration of information, we need only exhibit mechanisms by which information is brought together and exploited by later processes."   
**-David Chalmers, Facing Up to The Problem of Consciousness**

# Introduction

## Objective

General intelligence of the type we possess exists exclusively, for now, in the domain of the conscious human. Therefore, an understanding of the mechanisms leading to our conscious experience may be required if an artificial general intelligence is to one day be realized.

One defining aspect of human intelligence is our ability to subconsciously form new connections between abstract concepts, which then seem to "bubble up" to the forefront of our attention. This phenomenon, commonly called intuition, is responsible not only for our most startling and profound "Aha!" moments, but also for the seemingly arbitrary changes in our awareness of, say, the ticking of a clock on the wall. And although intuition is, unfortunately, a system that exists inside us as a “black box” (we have no conscious access to its decision-making process), the realization that we experience these shifts of attention unwillingly and "out of the blue" provides powerful clues to its underlying mechanisms.

Therefore, the primary purpose of this project is to develop an ensemble learning system (an *agent*) who's “conscious awareness” (its *attention*) receives input as connections between the symbolic elements of its environment as served up by its *intuition*. Further, noting the hierarchal nature of information, the agent was to be designed as one possible node in a larger network of such agents, facilitating the bootstrapping of an increasingly advanced intuition.

## Technologies

The framework and agent were developed in the Python (3.6) programming language, chosen for its rapid prototyping capabilities and reflective nature. 3rd party library usage includes KarooGP (genetic programming) and PyTorch (machine learning).

# Problem Domain

To facilitate exploration of this topic, a problem domain was chosen based on the realization that the agent’s environment should, ideally:

1. Afford the agent an opportunity to explore a complex, unknown search-space.
2. Be multi-context.
3. Provide mechanisms for signaling feedback to the agent.
4. Have the potential for practical application.

With that in mind, the agent was applied to the task of learning the Python programming language and, eventually, accomplishing some goal using it. This meets our criteria very well, because Python -

1. Is a complex space in which an isolated learning environment may be constructed.
2. Provides various contexts. Ex: keywords, functions, arguments, classes, instances, programs, etc.
3. Exposes methods such as keyword.iskeyword(s) and callable(s), enabling the agent to “ask” if some string s represents a Python keyword or a callable function, respectively. Further, because the agent itself is written in Python, custom methods may be written for the agent to provide more specific feedback, such as is\_python(s) which returns True if s represents a valid Python program with no syntax errors that generates no exceptions.
4. Is highly reflexive, allowing a sufficiently advanced agent to, quite literally, write a version of itself that solves some arbitrary problem. Indeed, even writing its own feedback mechanisms for querying fitness heuristics.

# Modeling Intuition

## Philosophy

In order to develop an agent capable of intuition, a model was designed based on the following observations regarding intuitive behavior in humans -

* OBSERVATION:   
  We often react to an event faster than we have time to logically decide on a rational course of action(CITE). Regardless, we are still able to make very good “in-the-moment” approximations that seem to come from outside the cognizant “I” inside each of us.  
    
  OBSERVATION:   
  We can articulate the rules we use to solve a given problem, but we are unable to explain why we chose to consider that set of rules over another (Pitrat, 2010).  
    
  OBSERVATION:   
  Shifts in changes to our awareness are often (if not always) autonomic (CITE). For example, we cannot dictate which song becomes stuck in our head, which clock we can suddenly hear ticking, or why we can “see” a circle   
    
  **CONCLUSION 1**:   
  Some system below the consciousness level exists for determining the information that gets brought to the forefront of our awareness.
* OBSERVATION:  
  Seemingly trivial and/or unproductive “mistakes” often come into our awareness. For example, songs DO get stuck in our head and we DO become suddenly aware of a ticking clock. Contrapositively, we’re often oblivious to important environmental queues, especially when properly distracted (a trait commonly exploited by magicians and pick pockets alike).  
    
  **CONCLUSION 2:**   
  Intuition is not perfect. However, "mistakes” have evolutionary value (e.g. genetic mutation compels biological evolution). Therefore, as a biological systems itself, it is reasonable to speculate that the mechanisms through which it learns to optimally processes and disseminate information are Darwinian in nature.  
  OBSERVATION: The contents of our awareness influences our intuition, and our intuition affects our awareness. awareness attenuates input from the intuition.  
    
  CONCLUSION: The two systems exist in a feedback loop. .. awareness is a working memory  
    
    
  **CONCLUSION 2**:
* OBSERVATION:   
  It is human nature to explore our environment, ourselves and our origins.  
    
  **CONCLUSION 3**:  
  An agent, left to its own devices, will naturally explore its environment.
* Intuition is (likely) not ephemeral. Complex systems in nature are generally feedback networks (CITE), and the brain and its myriad sub-systems appear to be no exception. Therefore, intuition and our conscioius awareness are likely a feedback network receiving it’s signals from both the current environmental context and our awareness of, our “concentration on”, those signals - it seems to guide us as much as we guide it. This is evident by the fact that by concentrating/focusing our awareness on a topic, we dictate the amount of work our sub-conscious performs on it – our awareness and intuition is a feedback loop
* A short-term, finite, “working memory” exists, allowing us to hold previous environmental symbols in our minds. The agent’s “memory” represents this and allows introduction of error into the connection-forming process.

## The Intuitive Model

The model consists of several signal channels (labeled) connecting 4 functional layers (labeled *Conceptual, Intuitive, Logical,* and *Output)* as described below and given by Figure 1.

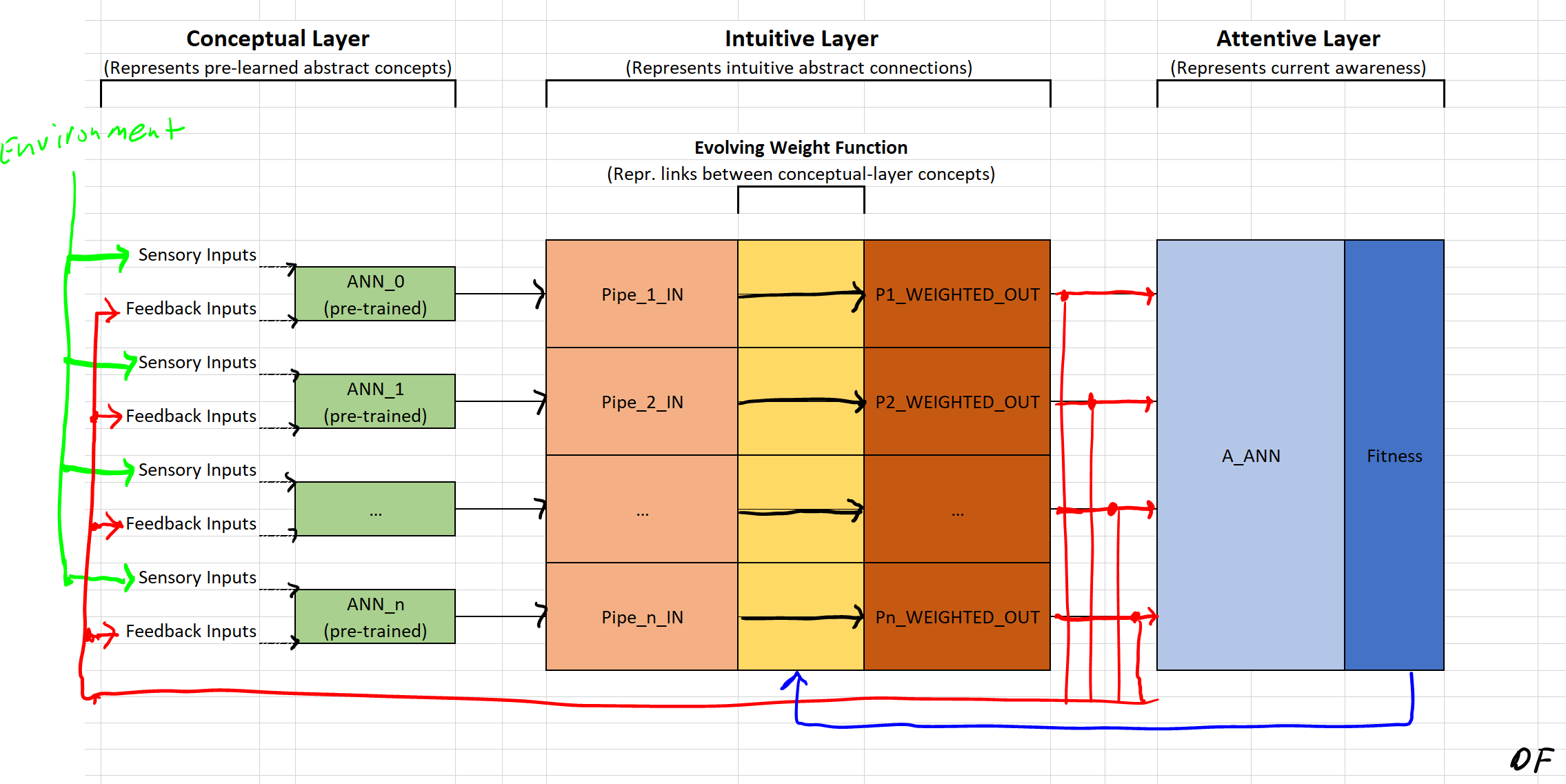


Figure 1

Signal channels   
Data is mostly **feed-forward**, with recurrent feedback signaling the agent's current contextual state and fitness.

**Note:** Several models were initially conceived (see: Failed Approaches/Models)

Note: Some layers are ephemeral, some are implemented in software.

### Layer 1 – The Conceptual Layer

**The conceptual layer represents** one's existing knowledge base (KB) of abstract concepts. It consists of a set `A` of artificial neural networks (ANNs) with the following properties -

Each ANN at this level has a set `X` of input nodes consisting of `k` feed-forward sensory input nodes and `m` feedback input nodes, defined as:

| X | Channel | Description |

|-------------|-------------| -------------|

| `x\_0` to `x\_k-1` | Feed-forward | Environmental sensory input, identical for every `a` in `A` |

| `x\_k` to `x\_k+m-1` | Feedback | Each `x\_k` to `x\_n` is mapped from a corresponding attentive-layer input node. This represents feedback of the agent's current context. |

Each ANN is pre-trained (offline), possibly each for a different class of objects (ex: `a\_0` classifies digits, `a\_1` classifies letters, etc). During this pre-training, each `x\_k` to `x\_m-1` input-value should be randomized to simulate environmental noise.

Each ANN's output nodes are provided to the intuitive layer as input.

* How is the genetic algorithm diff than just naother ann? Does it prevent overtraining

### Layer 2 -

**The intuitive layer represents** … and is a set of data pipes, one for each conceptual-layer ANN, connecting the conceptual layer to the attentive layer. On each state change, each pipe is weighted according to some fitness function that evolves in an online manner according to some genetic algorithm, who's fitness is received as feedback from the attentive layer.

These weights are subsequently used as a bias (possibly binary, logarithmic, etc) by the attentive layer. In this way, the agent's "intuition" learns how to best allocate the agent's "attention" while allowing "mistakes" to enter its awareness. These mistakes represent possible new connections between the conceptual layer's existing abstract concepts.

### Layer 3 – The Logical Layer

**The logical layer represents** .. .and examines the output-nodes of the intutive layer to draw concl

### Layer 4 – The Output Layer

**The output layer represents** …

# The Agent

* The agent exists in the context of its environmental inputs and “intuitively” learns the symbols present in it. In this way, it can be thought of as an automatic tokenizer.
* Context - an amalgamation of input from the five-senses input - we listen and see at the same time?

## The data

Extracted w/Pandas

Why this set?

* Familiar with it.
* Deep learnable but w/Simplicity - No need for conv or pooling of layers (though agent is capable of doing so).

Data Set desc: <https://archive.ics.uci.edu/ml/datasets/Letter+Recognition>

## Fitness Heuristics

* What heuristics might drive it and its exploration - neophilia? Self-actualization? Guilt?

## Agent operation/parameters/algorithms/methodology

“Step” – A Walkthrough

Tunable Constants…

# Performance

* Human context switching is approx 200 ms. 20-50ms is reasonably real-time.

# Failed Approaches/Models

…

# What’s Next

Bootstrapping:

* Each agent represents a context

Branching/Bootstrapping:

1. Branch into a hierarichal structure of agents (faciliated by reflection) – each representing one context. Contexts are not necessarily predefined.
2. L2 branch after some number of increases of the last x iters
3. Each agent represents a single "concept" such as a letter, or a python kwd
4. If a branch agent does not learn enough over some t, rm it (log to kb?) - it's inputs do not form any concepts
5. One node has one goal, a clique has another goal, a network has an even still more encompassing goal (ex: find free memory?)

L2 Tuning:

1. Monitor accuracy heuristics over time -
2. increase max pop size after some accuracy threshold
3. decrease max pop size if proprtion of unfit to fit outputs calls for it

More heuristics:

* We encounter many heuristics in life
* OBSERVATION: We are not always compelled to act rationally - in addition to logical reasoning, emotional (irrational) reasoning strongly influences our behavior.   
   **CONCLUSION 3** (Also in correlation w/CONCLUSION 2):  
  The heuristics guiding our intuition’s evolutionary journey may be associated with irrational drivers such as guilt, loneliness, boredom, etc.
  + Innate drive based on environmental queues/heuristics comples us toward the new, and
  + How are the inputs wighted? Biased by "amount" of input and log(type)? I.e. Fire vs. TV - Fire is big and new.
  + What drives the shifts?
  + The genetic alg decides attentive allocation between input data elements. Heuristics?

# Appendix A:

## Influences

Nicolai Tesla, Daniel Hofstadter, Ray Kurzweil, Richard Dawkins, Jacques Pitrat, Danial Kahneman

# References

Pitrat, J. (2010). Artificial beings Wiley-ISTE.

# Appendix B: Technical Information

## Usage Instructions

…

CLI –

Tunable constants -